

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-276139

(43)Date of publication of application : 06.10.2000

(51)Int.Cl.

G10H 1/00

(21)Application number : 11-078575

(71)Applicant : YAMAHA CORP

(22)Date of filing : 23.03.1999

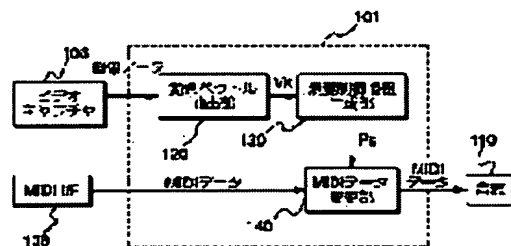
(72)Inventor : MIYAKI TSUTOMU

(54) METHOD FOR GENERATING MUSIC SOUND AND METHOD FOR CONTROLLING ELECTRONIC INSTRUMENT

(57)Abstract:

PROBLEM TO BE SOLVED: To control performance in accordance with an image through the use of a comparatively simple system.

SOLUTION: A movement vector extracting part 120 extracts a movement vector V_k based on image data which are outputted from a video camera and inputted with a video capture 106. A performance control information generating part 130 generates parameter control data P_s being performance control information based on the movement vector V_k which are extracted by the part 120. Besides, a MIDI data changing part 40 changes MIDI data which are outputted from an electronic musical instrument and inputted with a MIDI interface 108 based on parameter control data P_s which are generated by the part 130 and supplies the data to a sound source 110.



LEGAL STATUS

[Date of request for examination] 02.07.2002

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number] 3705000

[Date of registration] 05.08.2005

[Number of appeal against examiner's decision
of rejection]

[Date of requesting appeal against examiner's
decision of rejection]

[Date of extinction of right]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2000-276139

(P2000-276139A)

(43) 公開日 平成12年10月6日 (2000.10.6)

(51) Int.Cl.⁷

G 1 0 H 1/00

識別記号

1 0 2

F I

G 1 0 H 1/00

テーマコード(参考)

Z 5 D 3 7 8

1 0 2 Z

審査請求 未請求 請求項の数8 O L (全 13 頁)

(21) 出願番号

特願平11-78575

(22) 出願日

平成11年3月23日 (1999.3.23)

(71) 出願人 000004075

ヤマハ株式会社

静岡県浜松市中沢町10番1号

(72) 発明者 宮木 強

静岡県浜松市中沢町10番1号 ヤマハ株式会社内

(74) 代理人 100098084

弁理士 川▲崎▼ 研二 (外1名)

Fターム(参考) 5D378 KK01 KK34 KK50 MM12 MM14

MM42 MM47 MM48 MM64 MM65

MM68 MM92 QQ01 QQ25 TT18

TT22 UU31 UU41 XX05 XX27

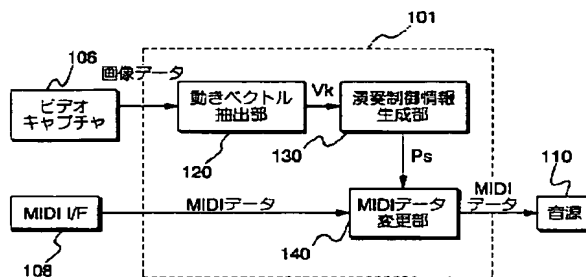
ZZ05

(54) 【発明の名称】 楽音生成方法および電子機器の制御方法

(57) 【要約】

【課題】 比較的簡易なシステムを用いて画像に応じた演奏の制御を行う。

【解決手段】 動きベクトル抽出部120は、ビデオカメラ200から出力され、ビデオキャプチャ106を介して入力される画像データに基づいて動きベクトルVkを抽出する。演奏制御情報生成部130は、動きベクトル抽出部120によって抽出された動きベクトルVkに基づいて、演奏制御情報であるパラメータ制御データPsを生成する。また、MIDIデータ変更部140は、演奏制御情報生成部130によって生成されたパラメータ制御データPsに基づいて、電子楽器400から出力されてMIDIインターフェイス108を介して入力されたMIDIデータの変更を行い、音源110に供給する。



【特許請求の範囲】

【請求項 1】 演奏の内容を示す演奏情報を供給する過程と、
 複数ブロックからなるビデオ画像を供給する過程と、
 供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、
 抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、
 算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、
 前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする楽音生成方法。

【請求項 2】 演奏情報を供給する過程と、
 前記演奏を制御する複数の演奏制御用パラメータのうちのいずれかを指定する過程と、
 複数ブロックからなるビデオ画像を供給する過程と、
 供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、
 抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、
 算出した前記制御ベクトルに基づいて、指定された前記演奏制御用パラメータを生成する過程と、
 前記演奏情報および生成された前記制御用パラメータに基づいて楽音を生成する過程とを備えることを特徴とする楽音生成方法。

【請求項 3】 演奏情報を供給する過程と、
 複数ブロックからなるビデオ画像を供給する過程と、
 供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、
 抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、
 前記制御ベクトルの時間的な変化態様に基づいて、前記演奏を制御する制御情報を生成する過程と、
 前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする楽音生成方法。

【請求項 4】 演奏の内容を示す演奏情報を供給する過程と、
 複数ブロックからなるビデオ画像を供給する過程と、
 供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、
 抽出した前記複数ブロックのうちの一部の領域に対応する複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、
 算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、
 前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする楽音生成方法。

【請求項 5】 演奏の内容を示す演奏情報を供給する過程と、

前記演奏を制御する複数の演奏制御用パラメータのうちのいずれかを指定する過程と、

複数ブロックからなるビデオ画像を供給する過程と、
 操作子の操作に応じて前記ビデオ画像の中の一部の領域を指定する過程と、

指定された前記一部の領域内の画像の変化を検出し、該変化を示す検出信号を発生する過程と、

10 発生した前記検出信号に基づいて、指定された前記演奏制御用パラメータを生成する過程と、

前記演奏情報および生成された前記制御用パラメータに基づいて楽音を生成する過程とを備えることを特徴とする楽音生成方法。

【請求項 6】 請求項 1 ないし 5 いずれかに記載の楽音生成方法において、

前記ビデオ画像の左右を反転させる反転過程と、

前記反転過程において出力されたビデオ画像を表示する表示過程とを備えることを特徴とする楽音生成方法。

20 【請求項 7】 演奏の内容を示す演奏情報を供給する過程と、

複数ブロックからなるビデオ画像を供給する過程と、
 供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、

抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、

算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、

30 前記演奏情報および生成された前記制御情報に基づいて電子機器を制御する過程とを備えることを特徴とする電子機器の制御方法。

【請求項 8】 請求項 7 に記載の電子機器の制御方法において、

前記ビデオ画像の左右を反転させる反転過程と、

前記反転過程において出力されたビデオ画像を表示する表示手段とを備えることを特徴とする電子機器の制御方法。

【発明の詳細な説明】

【0001】

40 【発明の属する技術分野】 この発明は、ビデオ画像を利用した楽音生成方法および電子機器の制御方法に関する。

【0002】

【従来の技術】 従来より、画像に対応して演奏の制御を行う技術が提案されている。このような技術としては、例えば特許第 2629740 号公報や、特許第 2629740 号公報に、被写体の輪郭を利用してテンポ等を制御する技術が開示されている。これらの技術によれば、入力されたビデオ信号から、R (赤) G (緑)、B (青) の各色信号を分離して、各色毎にデジタルデータ

として階調を表す階調データを生成する。そして、各色の階調データと予め定めたしきい値データとに基づいて被写体を特定し、当該被写体の輪郭を検出する。このように輪郭を特定した後、特許第 2629740 号公報に記載された技術では輪郭の複雑さに応じて演奏を制御し、特許第 2629740 号公報に記載された技術では被写体の重心の移動に応じて演奏を制御している。

【0003】

【発明が解決しようとする課題】しかしながら、上述した従来の技術においては、被写体を特定し輪郭を検出する必要があったので、処理にかかる負担が大きいという問題があった。すなわち、各画素について RGB 各色毎の階調データを作成してしきい値と比較しなければならず、その上さらに、検出した被写体の輪郭の複雑さや、重心の移動などを求めなくてはならなかった。また、被写体を特定するために予め定めるしきい値の決定も困難であり、必ずしも精度よく輪郭を検出できるとは限らなかった。このように、画像に基づいて演奏を制御するために従来から提案されている技術は、被写体検出のためのシステム構成が複雑になったり、しきい値を設定によって動作が不安定になるという問題があった。

【0004】本発明は、上述した課題を解決するためになされたものであり、比較的簡易なシステムを用いて、画像に応じた演奏の制御を容易に行うことができる楽音生成方法および電子機器の制御方法を提供することを目的としている。

【0005】

【課題を解決するための手段】上述した課題を解決するために、請求項 1 に記載の発明は、演奏の内容を示す演奏情報を供給する過程と、複数ブロックからなるビデオ画像を供給する過程と、供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする。また、請求項 2 に記載の発明は、演奏情報を供給する過程と、前記演奏を制御する複数の演奏制御用パラメータのうちのいずれかを指定する過程と、複数ブロックからなるビデオ画像を供給する過程と、供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、算出した前記制御ベクトルに基づいて、指定された前記演奏制御用パラメータを生成する過程と、前記演奏情報および生成された前記制御用パラメータに基づいて楽音を生成する過程とを備えることを特徴とする。また、請求項 3 に記載の発明は、演奏情報を供給する過程と、複数ブロックからなるビデオ画像を供給する過程と、供給

された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、前記制御ベクトルの時間的な変化態様に基づいて、前記演奏を制御する制御情報を生成する過程と、前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする。また、請求項 4 に記載の発明は、演奏の内容を示す演奏情報を供給する過程と、複数ブロックからなるビデオ画像を供給する過程と、供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、抽出した前記複数ブロックのうちの一部の領域に対応する複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、前記演奏情報および生成された前記制御情報に基づいて楽音を生成する過程とを備えることを特徴とする。また、請求項 5 に記載の発明は、演奏の内容を示す演奏情報を供給する過程と、前記演奏を制御する複数の演奏制御用パラメータのうちのいずれかを指定する過程と、複数ブロックからなるビデオ画像を供給する過程と、操作子の操作に応じて前記ビデオ画像の中の一部の領域を指定する過程と、指定された前記一部の領域内の画像の変化を検出し、該変化を示す検出信号を発生する過程と、発生した前記検出信号に基づいて、指定された前記演奏制御用パラメータを生成する過程と、前記演奏情報および生成された前記制御用パラメータに基づいて楽音を生成する過程とを備えることを特徴とする。また、請求項 6 に記載の発明は、請求項 1 ないし 5 いずれかに記載の楽音生成方法において、前記ビデオ画像の左右を反転させる反転過程と、前記反転過程において出力されたビデオ画像を表示する表示手段とを備えることを特徴とする。また、請求項 7 に記載の発明は、演奏の内容を示す演奏情報を供給する過程と、複数ブロックからなるビデオ画像を供給する過程と、供給された前記ビデオ画像から前記複数ブロックに対応する複数の動きベクトルを抽出する過程と、抽出した前記複数の動きベクトルから 1 つの制御ベクトルを算出する過程と、算出した前記制御ベクトルに基づいて、前記演奏を制御する制御情報を生成する過程と、前記演奏情報および生成された前記制御情報に基づいて電子機器を制御する過程とを備えることを特徴とする。また、請求項 8 に記載の発明は、請求項 7 に記載の電子機器の制御方法において、前記ビデオ画像の左右を反転させる反転過程と、前記反転過程において出力されたビデオ画像を表示する表示手段とを備えることを特徴とする。

【0006】

【発明の実施の形態】以下、図面を参照しながら、本発明の実施の形態について説明する。

【0007】1. 実施形態の構成

1-1. 実施形態の概要構成

図1は、実施形態の概要構成を示す図である。本実施形態は、図1に示すように、本発明にかかる演奏制御装置100に、ビデオカメラ200、ディスプレイ300、電子楽器400、操作子500、およびサウンドシステム600が接続された構成となっている。演奏制御装置100は、ビデオカメラ200によって撮影された画像および操作子500の操作に基づいて、電子楽器400から入力する演奏あるいは演奏制御装置100における自動演奏を制御する装置であり、後に図2を参照して説明する構成を備えている。ビデオカメラ200は、被写体である演奏者の動作を撮影し、ビデオ信号を生成して出力する装置である。ディスプレイ300は、ビデオカメラ200によって撮影された画像を表示する装置であり、例えばCRT (Cathode Ray Tube)、LCD (Liquid Crystal Display)、プラズマディスプレイなどを用いる。なお、本実施形態では640画素×480画素の解像度のディスプレイを用いて説明するが、解像度はこれに限定されるものではない。

【0008】電子楽器400は、演奏操作子(鍵盤やペダルなど)の操作状態に応じてMIDI (Musical Instrument Digital Interface) 規格に基づいた演奏データを生成して出力する装置であり、例えば電子ピアノやMIDIギターなどを用いる。操作子500は、演奏制御装置100における種々の設定を行うためのものであり、例えばキースイッチやパネル操作子などが該当する。この操作子500は、演奏制御装置100に備え付けられていてもよいし、別体としてケーブル接続されていてもよい。また、赤外線や電波などの無線によって操作状態を送信できるようになっていてもよい。そして、サウンドシステム600は、アンプやスピーカなどから構成されており、演奏制御装置100から出力される音響信号を増幅して放音する装置である。

【0009】1-2. 演奏制御装置100

次に、図2を参照しながら、演奏制御装置100の構成について説明する。図2に示すように、演奏制御装置100は、バスを介して接続されたCPU101、ROM102、RAM103、ハードディスク104、タイマ105、ビデオキャプチャ106、ビデオ再生107、MIDIインターフェイス108、操作子インターフェイス109、および音源110を備えて構成されている。CPU101は、ROM102に記憶された各種プログラムに基づいて、バスを介して接続された各部の制御を行う。ROM102には、演奏制御装置100の動作を制御する各種プログラムが記憶されている。RAM103は、CPU101の動作に必要なワークエリアが設定される他、データを一時的に記憶するために用いられる。ハードディスク104は、大容量のデータを記憶できる大容量記憶媒体であり、演奏制御装置100において自動演奏を行う場合に用いる曲データなどが記憶さ

れる。

【0010】次に、タイマ105は、時間を計時して現在時刻を示すものである。ビデオキャプチャ106は、ビデオカメラ200から出力されるビデオ信号をデジタルデータである画像データに変換するものである。ビデオ再生107は、画像データをビデオ信号に変換してディスプレイ300に出力するものであり、例えばビデオメモリやビデオコントローラなどを備えたボードが該当する。また、MIDIインターフェイス108は、電子楽器400とのインターフェイス動作を行い、操作子インターフェイス109は、操作子500とのインターフェイス動作を行う。そして、音源110は、MIDIデータに基づいて楽音信号を生成するものであり、生成した楽音信号は、サウンドシステム600から放音される。

【0011】2. 実施形態の動作

次に、上記構成を有する実施形態の動作について説明する。

【0012】2-1. 概要処理

まず、本実施形態における処理の概要について説明する。

【0013】(1) 動作モード

本実施形態においては、予め設定された動作モードの中から、所望する動作モードを選択できるようになっている。動作モードには、

①電子楽器400からリアルタイム入力されたMIDIデータに所定の処理を施すことによって演奏制御を行う「外部入力モード」、

②ハードディスク104に記憶されたMIDI曲データを読み出して、MIDIデータシーケンスデータを生成することによって演奏制御を行う「シーケンサモード」、および、

③演奏者を左右反転させた画像をディスプレイ300に表示して、演奏者がディスプレイ300を見ながら動作することによって演奏制御を指示する「反転画像モード」、の3種類のモードがある。ここで、図3、図4および図5は、各動作モードにおける演奏制御装置100の動作の概要を示す機能ブロック図であり、図3は、外部入力モードにおける概要動作を、図4はシーケンサモードにおける概要動作を、図5は反転画像モードにおける概要動作をそれぞれ示している。以下、外部入力モード(図3)、シーケンサモード(図4)、反転画像モード(図5)の動作概要についてそれぞれ説明する。

【0014】①外部入力モード

外部入力モードにおいては、図3に示すように、ROM102に記憶されたプログラムに基づいたCPU101の処理によって、動きベクトル抽出部120、演奏制御情報生成部130、およびMIDIデータ変更部140が構成される。動きベクトル抽出部120は、ビデオカメラ200から出力されビデオキャプチャ106を介し

て入力された画像データに基づいて動きベクトル V_k を抽出する。なお、動きベクトル V_k については、後に詳しく説明する。演奏制御情報生成部130は、動きベクトル抽出部120によって抽出された動きベクトル V_k に基づいて、演奏制御情報であるパラメータ制御データ P_s を生成する。また、MIDIデータ変更部140は、演奏制御情報生成部130によって生成されたパラメータ制御データ P_s に基づいて、電子楽器400から出力されてMIDIインターフェイス108を介して入力されたMIDIデータの変更や追加を行い、音源110に供給する。ここで変更するMIDIデータの変更、追加とは、例えば、コントロールチェンジメッセージで制御されている音量、パン、EGパラメータ等を、その現在値に対する相対値ないし絶対値として制御する新たなメッセージを発生したり、ノートオンメッセージに含まれる音高やベロシティ値を変更したり、シーケンサ等を制御するシステムメッセージを発生する等のあらゆるMIDIデータの制御を含む。例えば、電子楽器400から出力されたMIDIデータが、ボリュームを制御するコントロールチェンジメッセージであれば、MIDIデータ変更部140は、コントロールチェンジメッセージ中に含まれるボリュームを示す値をパラメータ制御データ P_s に基づいて変更する。

【0015】②シーケンサモード

シーケンサモードにおいては、図4に示すように、ROM102に記憶されたプログラムに基づいたCPU101の処理によって、動きベクトル抽出部120、演奏制御情報生成部130、およびシーケンサ150が構成される。なお、動きベクトル抽出部120および演奏制御情報生成部130は、図3に示したものと同様である。シーケンサ150は、ハードディスク104から読み出されRAM103に記憶された曲データを解釈し、MIDIデータを生成して音源110に供給する。ここで、シーケンサ150は、演奏制御情報生成部130によって生成されたパラメータ制御データ P_s に基づいて、曲データに基づいて生成すべきMIDIデータを変更したり、曲データに基づいて生成されたMIDIデータを変更、追加する。例えば、曲データによって指示されるテンポが120である場合に、パラメータ制御データ P_s がテンポアップを指示しているときは、シーケンサ150はテンポが例えば150になるようにMIDIデータを音源110に出力する。あるいは、曲データに基づいて生成されたMIDIデータを、①の場合と同様に変更ないし追加する。

【0016】③反転画像モード

反転画像モードにおいては、図5に示すように、ROM102に記憶されたプログラムに基づいたCPU101の処理によって、動きベクトル抽出部120、演奏制御情報生成部130、MIDIデータ変更部140および画像反転部160が構成される。なお、動きベクトル抽

出部120、演奏制御情報生成部130、およびMIDIデータ変更部140は、図3に示したものと同様である。画像反転部160は、ディスプレイ300に表示させるべき画像を左右反転させるための画像データを生成する。これにより生成された画像データに基づいてディスプレイ300に表示される画像は、鏡に映った画像のようになる。この動作モードにおいて適した演奏制御処理については、後述する(図13参照)。なお、電子楽器400を接続せずに、MIDI曲データを用いる場合は、MIDIインターフェイス108はハードディスク104およびRAM103に、MIDIデータ変更部140はシーケンサ150に置き換わる。

【0017】(2)動きベクトル抽出部

次に、動きベクトル抽出部120について説明する。動きベクトル抽出部120は、ビデオキャプチャ106から連続して入力される複数フレームの画像データに基づいて、以下に説明するように動きベクトル V_k を検出する。なお、「動きベクトル」は、各フレーム画面における動きの大きさと方向を示すデータであり、画像データやオーディオデータをデータ圧縮する国際標準規格であるMPEG (Moving Picture Experts Group phase) における動き補償に用いられている。本実施形態において、「動きベクトル」は、ある時点(k フレーム)における画素の位置を、過去($k-1$ フレーム)の画素を用いて表される。ここで、1フレーム分の画像データに含まれる個々の座標のデータを画素と呼ぶ。具体的には、 k フレームの画素 X_{ijk} (ij は座標、 k はフレーム)を示すため、 $k-1$ フレームにおける画像の中から X_{il} に最も近い画素値 X_{im} を検索する。そして、 X_{il} と X_{im} との空間的なずれ $|i-l|$ と $|j-m|$ を求める。 X_{il} に最も近い画素値 X_{im} とは、水平および垂直に15画素範囲内ですらして、もっとも差分が小さな画素をいう。本実施形態では、1フレーム画面を構成する複数の画素を、16画素×16画素を1画素ブロックとする複数の画素ブロックに分けて扱い、各画素ブロックの動きベクトルを求めて、これらの動きベクトルを総合的に示したものを、 k フレームにおける1画面分の動きベクトル V_k とする。フレーム k における画素ブロックの動きベクトルは、具体的には次のように求められる。

【数1】

$$\sum_{l,j=0}^{15} |x_{i,j,k} - x_{l \pm a, j \pm b, k-1}|$$

a:0~15
b:0~15

数1において最小となる画素位置(a, b)の値をこの画素ブロックの動きベクトルとする。

【0018】より具体的には、パラメータ a, b にそれぞれ0~15の値を与えながら、フレーム k における X_l

1〜X15 15までの各画素の画素値と、フレームk-1における $X1 \pm a1 \pm b \sim X15 \pm a15 \pm b$ との差の絶対値を累算した場合に、数1の解が最小となる(a, b)の組を検索する。そして、画素位置の差が(a, b)である各画素値同士を比較して画素値の差がなければ、フレームk-1からフレームkまでの間に注目画素は(a, b)だけ移動したことになる。そして、画素ブロックに含まれる全画素について画素値の差を求め、この差を累算した値が最小となる(a, b)が、この画素フレームの動きを最もよく表す各画素ブロックの動きベクトルとして検出される。但し、前記差を累算した値が最小となる(a, b)について、その累算値が所定のしきい値以下にならない場合には、その画素ブロックの動きベクトルは検出不能と判断される。図6は、各画素ブロックにおいて検出した(a, b)を画素ブロックの中心からのベクトルとして示している。本実施形態では1画面の画素数は640画素×480画素であるから、縦30×横40個の画素ブロックの動きベクトル $Vk1 \sim Vk1200$ が検出される。1画面分の動きベクトル Vk は、例えば図7に示すように、各画素ブロックの動きベクトル $Vk1 \sim Vk1200$ の平均値、重み付け平均値、自乗平均値等として算出されたベクトルとしたり、複数画素ブロックの複数ベクトルの中から選択された最大の大きさをもつベクトルとすればよい。あるいは、該大きさが上位のn個のベクトルの平均ベクトルが最も大きくなるものの平均ベクトルとしてもよい。このように、「動きベクトル」を抽出すれば、被写体を特定する必要なく動き量を示すことができるようになる。

【0019】(3) 演奏制御生成部

次に、演奏制御情報生成部130の動作について説明する。演奏制御情報生成部130は、動きベクトル抽出部120において抽出された動きベクトル Vk に基づいて、演奏のパラメータ制御データ Ps を生成して出力する。本実施形態では、パラメータ制御データ Ps を生成する処理として、

①操作子500を操作することによって、いずれのパラメータについて制御を行うかを選択することができる

「操作子選択モード」、

②指揮者が腕を上下に振る動作に基づいてテンポを制御する「テンポモード」、

③予め設定されたエリア内の動きベクトルに基づいて制御を行う「エリアモード」、

の3つの制御モードの中から選択できる。なお、各モードはパラメータ制御データ Ps を生成する処理の例示であって、以下に説明する各態様に限定されるものではなく、3つの制御モード中に示す各要素を任意に組み合わせたモードであっても構わない。

【0020】①操作子選択モード

操作子選択モードは、操作子500を操作することによって、いずれのパラメータについて制御を行うかを選択

することができるモードである。ここで、操作子500は、例えば演奏者が手に装着する「指スイッチ」であり、この「指スイッチ」は、人指し指で操作するスイッチ Sh と、中指で操作するスイッチ Sn とを備えており、いずれかのスイッチを親指で押下することによってオン状態にすることができるようになっている。本実施形態では、人指し指で操作するスイッチ Sh がオン状態の場合は、ボリューム制御用のパラメータ制御データ VOL を動きベクトルに基づいて生成する。また、中指で操作するスイッチ Sn がオン状態の場合は、テンポパラメータ制御用のパラメータ制御データ ΔTMP を動きベクトルに基づいて生成する。スイッチ Sh 、 Sn のいずれもがオフ状態の場合は、パラメータ制御を行わず、パラメータ制御データ Ps は生成しない。なお、このモードにおける具体的な処理は、図11を参照しながら後述する。

【0021】②テンポモード

テンポモードは、指揮者が腕を上下に振る動作に基づいてテンポを制御するモードであり、上下方向の動きベクトルを検出してテンポ制御用のパラメータ制御データ Pst を生成する。なお、このモードにおける具体的な処理は、図12を参照しながら後述する。

【0022】③エリアモード

エリアモードは、予め設定されたエリア内の動きベクトルに基づいて制御を行うモードである。本実施形態では、図8に示すように、3つのエリア($i1$ 、 $i2$ 、 $i3$)が予め設定されており、各エリアにおける動きベクトルを検出して、それぞれのエリアに割り当てられたパラメータの制御を行う。それぞれに割り当てられたパラメータは、例えばエリア $i1$ はボリューム、エリア $i2$ はテンポ、エリア $i3$ はパンニングである。

【0023】また、ディスプレイ300には、図9に示すように被写体を左右反転させて表示されるようになっており(上述した反転画像モード)、演奏者にとっては、鏡に映った自分の姿を見ながら動作できるようになっている。そして、予め各パラメータに割り当てられたエリア $i1$ 、 $i2$ 、 $i3$ は、演奏者が確認できるように、ディスプレイ300にスーパーインポーズによって表示される。例えば、演奏者がボリュームの制御を所望する場合は、ディスプレイ300に表示されたエリア $i1$ 内に自己の動作が表示されるようにすれば、演奏制御装置100がエリア $i1$ 内における動きベクトルを検出してボリュームの制御を行う。なお、このモードにおける具体的な処理は、図13を参照しながら後述する。

【0024】2-2. 処理の内容

次に、上述した本実施形態における動作を実現するためのCPU101にける処理について、フローチャートを参照(10図〜13図)しながら説明する。

【0025】(1) メインルーチン

まず、図10は、メインルーチンを示すフローチャート

である。メインルーチンが起動され、処理が開始すると、まず、CPU101は初期設定を行う(S101)。初期設定においては、例えば操作子500の状態などに基づいて動作モードを選択する処理や、各種変数の初期化などを行う。なお、画像データをビデオキャプチャ106から取り込んで、RAM103に記憶しておく処理は、キャプチャ106によって、以下に説明するステップS102～S105のCPU処理と並行して行われており、後述するように、ステップS106において読み出され、動きベクトルが抽出されるものとする。CPU101は、初期設定を終了すると、次に、MIDI処理を行う(S102)。MIDI処理は、ステップS101の初期化処理において外部入力モード(図3)あるいは反転画像モード(図5)が選択された場合には、電子楽器400から出力されるMIDIデータを読み込む処理であり、シーケンサモード(図4)が選択された場合には、シーケンサ150において曲データに基づいてMIDIデータを生成する処理である。

【0026】CPU101は、MIDI処理に続いて、操作子処理を行う(S103)。操作子処理は、操作子500の状態を読み込んでRAM103に記憶したり、状態の変化に応じて各種設定を変更する処理などが該当する。操作子処理が終了すると、CPU101は、その他の処理を行う(S104)。その他の処理とは、ステップS102において読み込んだないし生成したMIDIデータに基づいて、音源110における楽音の生成を制御したり、ステップS103において読み込まれた操作子500の状態に応じたシステム状態の変更などを行う処理である。

【0027】その他の処理が終了すると、CPU101は、所定のタイミングから所定の時間が経過したか否かをタイマ105によって計時された時刻に基づいて判別する(S105)。ここで、所定のタイミングとは、当該判別が最初の判別であれば、初期設定(S101)を行ったタイミングをいい、当該判別が2回目以降の判別であれば、前回動きベクトルの抽出(S106)を行ったタイミングをいう。ステップS105の判別において、所定の時間が経過していないと判別した場合は(S105; NO)、処理をステップS102に移行させ、再びMIDI処理を行う。すなわち、タイマ105が所定の時間を計時するまでの間は、ステップS102～ステップS105を循環しながら、動きベクトルに影響されない楽音生成処理を行っている。

【0028】一方、ステップS105の判別において、所定の時間が経過していると判別した場合は(S105; YES)、所定時間内に入力された画像データに基づいて前述したように動きベクトルを抽出する処理を行う(S106)。そして、ステップS106において抽出された動きベクトルに基づいて、動きベクトルに応じた楽音制御処理を行う(S200)。この楽音制御処理

は、前述した各モード(①操作子選択モード、②テンポモード、③エリアモード)に応じて異なるので、以下、楽音制御処理について、各モード毎に処理を説明する。

【0029】(2) 楽音制御処理

①操作子選択モード

まず、操作子選択モードにおける処理について図11に示すフローチャートを参照しながら説明する。楽音制御処理が開始すると、まず、メインルーチンのステップS103において読み込んだ操作子500の状態についての判別を行う(S2101)。操作子500には、前述の指スイッチが含まれており、ここでは、人指し指で操作するスイッチShがオン状態の場合はボリューム制御用のパラメータ制御データVOLを生成し、中指で操作するスイッチSnがオン状態の場合はテンポ制御用のパラメータ制御データΔTMPを生成する。スイッチSh、Snのいずれもがオフ状態の場合は、パラメータ制御を行わず、パラメータ制御データPsは生成しない。従って、ステップS2101の判別において、スイッチShがオン状態であると判別した場合は、音量を制御するパラメータ制御データVOL(MIDIデータの1つ)を生成する処理に移行する(S2102)。具体的には、ステップS106において抽出した時刻kにおけるフレームkの各動きベクトルV_{k1}～V_{k1200}に基づいて、各動きベクトルのy方向の成分を求め、1画面分の全画素ブロックのy方向成分の平均値V_kを検出する。そして、V_kに基づいてパラメータ制御データVOLを生成する。そして、パラメータ制御データVOLに応じて楽音の音量を制御する(S2103)。具体的には、その時点の音領地をパラメータ制御データVOLに基づいて決定された音量値をパラメータとするMIDIデータ(例えばコントロールチェンジメッセージ)を発生する。なお、過去に発生した音量を制御するMIDIデータの値を保持しておけば、その時点の音量値は判別できる。このように、抽出された動きベクトルのy方向成分の平均値に応じて音量を制御したのち、処理をメインルーチンに戻す。

【0030】ところで、ステップS2101の判別において、スイッチSnがオン状態であると判別した場合は、テンポを制御するパラメータ制御データΔTMPを生成する処理に移行する(S2104)。具体的には、ステップS106において抽出した時刻kにおけるフレームkの各動きベクトルV_{k1}～V_{k1200}に基づいて、各動きベクトルのうちの最大ベクトルの長さを検出して、パラメータ制御データΔTMPを生成する。そして、パラメータ制御データΔTMPに応じて楽音のテンポを制御する(S2105)。具体的には、例えば、曲データが指示するテンポが120であれば、本来は4分音符1拍分のノートオンメッセージを0.5秒間隔で出力する(1分間に4分音符が120拍)が、パラメータ制御データΔTMPがテンポを60まで下げることを指

示していた場合は、シーケンサ150は、4分音符1拍分のノートオンメッセージを1秒間隔で出力する（1分間に4分音符が60拍）。このように、動きベクトルの最大ベクトルの長さに応じてテンポを制御した後、処理をメインルーチンに戻す。ところで、ステップS2101の判別において、スイッチShあるいはスイッチSnのいずれかがオフ状態であると判別した場合は、楽音制御処理を行わず、そのまま処理をメインルーチンに戻す。

【0031】このように、操作子500を操作することによって、動きベクトルをどのような制御に用いるかを選択できるので、演奏中に制御内容を自由に切り換えることができるようになる。

【0032】②テンポモード

次に、テンポモードにおける処理について図12に示すフローチャートを参照しながら説明する。テンポモードは、動きベクトルのy方向成分に基づいてテンポを制御するモードであるので、まず、y方向成分のピークを検出して変数Vyの値とする（S2201）。次に、ステップS2201において検出した値をローパスフィルタに通して雑音と除去し、変数Vyとする（S2202）。テンポモードは、y方向における被写体の往復運動の速さに基づいてテンポを制御するので、y正方向の動きが加速から減速に転じたタイミングを検出し、前回の加速から減速に転じたタイミングとの時間間隔で1往復運動に要した時間を計測する処理を行う。具体的には、y方向における被写体の動きは、だんだん加速してピークを迎えた後減速し、戻りの動作が行われるという1サイクルを仮定し、以下に説明するように、動きベクトルのy成分であるVyの大きさを、y方向における被写体の動作の速さとを示す値として扱う。また、1サイクルにおける動きベクトルのy成分の最大値Pyを、y方向における被写体の動作のピークを示す値として扱う。そして、y正方向における演奏者の動作状態を、ST=0、ST=1、ST=2のいずれかによって示すものとしている。ここで、ST=0は、被写体のy方向における動きが検出されない場合を示している。この場合は、ステップS2202で決定したVyは、負の値を示しているか、後述するように、しきい値Tyを超えていない。また、ST=1は、被写体のy方向における動きがy正方向において加速している場合を示している。この場合は、ステップS2202で決定したVyは、しきい値Tyを超えている。また、Vyは、前回更新した最大値Pyを超えており、被写体のy方向における動きがさらに速くなったことを示している。そして、ST=2は、被写体のy方向における動きが正方向において減速している場合を示している。この場合は、ステップS2202で決定したVyは、正の値を示すが、最大値Py以下となる。

【0033】このような規則に従って被写体の動き状態

を判別するために、ステップS2202においてVyを決定した後、動きベクトルのy方向成分の態様を示すステートメントSTを判別する（S2203）。まず、ステップS2203の判別において、ST=0、すなわち正方向の動きが検出されていないと判別した場合は（S2203；ST=0）、次に、 $Vy > Ty$ 、すなわちy方向のピーク値Vyがしきい値Tyよりも大きいかなかを判別する（S2204）。ここで $Vy \leq Ty$ と判別した場合は（S2204；NO）、検出されたy方向のピーク値がしきい値以下（負方向を含む）であるので、ステートメントSTの値を変更せず、そのまま処理をメインルーチンに戻す。

【0034】一方、ステップS2204の判別において $Vy > Ty$ であると判別した場合は（S2204；YES）、正方向への動きを検出したのでステートメントSTを1とし、ステップS2201において検出したVyを当該サイクルのピーク値Pyとして（S2205）、処理をメインルーチンに戻す。すなわち、次のステップS2203の判別においてST=1と判別すると（S2203；ST=1）、次に、 $Vy > Py$ であるかなかを判別を行う（S2206）。ここでは、S2201において検出されたVyが当該サイクルのピーク値Pyよりも大きければ、被写体の動きは未だ加速中であると判別できる。従って、 $Vy > Py$ であると判別した場合は（S2206；YES）、PyをVyで更新して（S2207）、処理をメインルーチンに戻す。

【0035】一方、ステップS2206の判別において、 $Vy \leq Py$ と判別した場合は（S2206；NO）、y方向成分がピーク値Pyを下回り始めたことと判別できるので、ステートメントSTを2とし、現在時刻を示す変数tを、タイマ105（図2参照）によって計時している時刻とする（S2208）。そして、ステップS2208における時刻を示すtから同様の処理を前回行った際の時刻を示すt0を引いて、1サイクルに要した時間Δtを求める。Δtを求めた後は、次のサイクルにおいて同様に時間差を算出できるように、t0の値をtの値で更新する（S2209）。このようにして求めた1サイクルに要した時間Δtに応じて、例えば、Δtが現在のテンポ値に対応した拍の時間間隔に比べて小さければテンポを加速し、大きければテンポを減速するといったテンポ制御を行い（S2210）、処理をメインルーチンに戻す。ただし、Δtと前記拍の時間間隔の差が大きく離れていた場合には、前記テンポの制御は行わない方がよい。また、拍の時間間隔と比較する代わりに、2拍の時間間隔や1小節の時間間隔と比較するようにしてもよい。

【0036】被写体のy方向への動きが減速に転じてステートメントST=2となると、ステップS2203の判別の後、 $Vy < 0$ であるかなかを、すなわち、動きベクトルのy成分の値が正に転じたかなかを判別し（S22

1.1)、 $V_y \geq 0$ と判別すると(S2211; NO)、被写体のy正方向における動きは減速中であるので、次のサイクルに移行していない。従って、ステートメントSTの値は変更せず、そのまま処理をメインルーチンに戻す。一方、ステップS2211の判別において、 $V_y < 0$ であると判別した場合は(S2211; YES)、被写体のy方向における動作が負方向に転じたので、次のサイクルに移行させるためにステートメントST=0とし(S2212)、処理をメインルーチンに戻す。

【0037】このような処理を循環する過程において、演奏者が上下に手を振る動きに応じてST=0→ST=1→ST=2→ST=0……というサイクルが形成され、ST=1からSR=2に転じる時刻において、前回ST=1からSR=2に転じた時刻との時間差 Δt に応じたテンポ制御が行われる。従って、被写体である演奏者が手を速く動かせばテンポが速くなり、遅く動かせばテンポが遅くなる。ここでは、複数画素ブロックの動きベクトルからピーク値を抽出してテンポ制御を行ったが、ピーク値の代わりに平均値等を使用してもよい。

【0038】③エリアモード

次に、エリアモードにおける処理について図13に示すフローチャートを参照しながら説明する。本実施形態においては、図8に示したように、予め設定されたエリアi1、i2、およびi3における動きベクトルに基づいて制御を行うので、各エリアについて順次動きベクトルの抽出を行い、当該エリアに割り当てられたパラメータについて動きベクトルに基づいた制御を行う。そこで、まずエリアi1から処理を行うべく、エリアを示す変数iを1に設定し(S2301)、画面中のエリアi1の最大ベクトルを抽出し、x方向成分とy方向成分を(M_x , M_y)とする(S2302)。なお、(M_x , M_y)の具体例については、図9に示している。そして、まず動きベクトルのy成分を示す M_y を使用するか否かを判別し(S2303)、 M_y を使用すると判別した場合は(S2303; YES)、次に M_y に応じてパラメータPYiを制御する(S2304)。パラメータPYiとは、エリアiにおけるy方向に割り当てられた制御対象のパラメータである。本実施形態では、エリアi1にはボリュームが割り当てられており、動きベクトルのy方向成分に基づいてボリュームを制御する。従ってi=1である場合は、ステップS2303の判別においてYESと判別されて、y方向のベクトル成分 M_y に応じてボリュームが制御される。

【0039】ステップS2303の判別において、 M_y を使用しないと判別した場合(S2303; NO)、あるいは、ステップS2304の処理を終了した場合は、次に、動きベクトルのx成分を示す M_x を使用するか否かを判別し(S2305)、 M_x を使用すると判別した場合は(S2305; YES)、 M_x に応じてパラメータPXiを制御する(S2306)。ここで、パラメー

タPXiは、エリアiにおけるx方向に割り当てられた制御対象のパラメータである。本実施形態では、エリアi3にはパンニングが割り当てられており、動きベクトルのx方向成分に基づいてパンニングを制御する。従ってi=3である場合は、ステップS2305の判別においてYESと判別されて、x方向のベクトル成分 M_x に応じてパンニングが制御され、左右のバランスが変化する。

【0040】ステップS2305の判別において、 M_x を使用しないと判別した場合(S2305; NO)、あるいは、ステップS2304の処理を終了した場合は、次に、残りのエリアがあるか否かを判別する(S2307)。すなわち、予め設定されたエリアi1、i2、i3の全てについて処理を行ったか否かを判別する。ステップS2307の判別において、残りエリアがあると判別した場合は(S2307; YES)、次のエリアの処理を行うべく変数iを1インクリメントして(S2308)、処理をステップS2302に移行させる。一方、残りエリアがない、すなわちエリアi1、i2、i3の全てについて処理が終了したと判別した場合は(S2307; NO)、処理をメインルーチンに戻す。

【0041】このように、ステップS2302からS2307の処理を循環する過程において、エリアi1においては、y方向のベクトル成分に応じてボリュームが制御(S2304)、エリアi2においては、y方向のベクトル成分に応じてテンポが制御され(S2304)、エリアi3においては、x方向のベクトル成分に応じてパンニングが制御される(S2306)。なお、設定するエリアの数や形状、また、そのエリアを画面上の何処に配置するかは任意であり、例えば、操作子500の操作に応じてユーザが任意に選択・設定できるようにすればよい。さらに、各エリアの動きベクトルを何で制御するかも任意であり、同じく、操作子500の操作に応じてユーザが任意に設定・選択できるようにすればよい。また、上記説明では、1つのエリアの動きベクトルのx成分とy成分のいずれか一方でパラメータを制御するようにしたが、xy両方の成分で同一ないし互いに異なるパラメータを独立して制御するようにしてもよい。

【0042】2-3. まとめ

このように、ビデオカメラ200で撮影した画像に基づいて、まず画像全体の各画素ブロックの動きベクトルを機械的に求め、その1画面分の複数の動きベクトルから制御用の動きベクトルを生成して楽音制御を行うので、被写体における検出すべき映像要素の位置を正確に特定する必要がなくなり、動きに応じた制御を容易に行うことができるようになる。また、画素ブロック毎の動きベクトルを抽出する技術は、MPEGなどの規格で、すでに広く用いられているので、その技術の進歩を容易に本発明に組み入れていくことができる。すなわち、本発明によれば、画像圧縮の目的で抽出される動きベクトル

を、演奏の制御に容易に用いることができるようになる。

【0043】3. 変形例

なお、本発明は既述した実施形態に限定されるものではなく、以下のような各種の変形が可能である。

【0044】動画像から抽出された動きベクトルに基づいて制御量を決定する手段は、上記実施形態に限定されるものではなく、例えば、上記実施形態では、ビデオカメラ200によって撮影されたビデオ信号に基づいて動きベクトルを抽出しているが、MPEGの画像データ等に含まれている画素ブロック毎の動きベクトルを利用してもよい。また、動きベクトルを抽出する際の画素ブロックの単位は、上記実施形態のように16画素×16画素に限らず、例えば8画素×8画素、5画素×10画素、その他3角形の形状等、任意の画素ブロックでもかまわない。また、上記実施形態では、例えば操作子選択モード(図11参照)ではY方向の成分だけを使用して制御しているが、ここで、x成分を用いて制御するようにしても構わなく、y方向の成分とx方向の成分との両方を使用して制御を行っても構わない。同様に、他の例(図12、図13)においても、使用する成分は実施例に限定されるものではない。また、必ずしもフレームごとに動きベクトルを抽出する必要はなく、複数フレームおきに抽出するようにしてもよい。

【0045】また、上記実施形態においてフレームkにおける動きベクトル V_k は、各画素ブロックの動きベクトルの平均値として説明しているが、これに限らず、全画素ブロックの動きベクトルのピーク値や最大値など、他の代表値でも構わない。動作モードや制御モードも上記実施形態に限定されるものではなく、例えば、一定時間内の複数フレームにおける動きベクトルを積分して作成した、積分画像を用いて制御するようなモードがあってもよい。また、上記実施形態の外部入力モード(①)においては、動きベクトルにより制御する制御対象を、操作子500に設けられた人指し指と中指のスイッチで選択するものとして説明したが、選択の方法はこれに限らない。例えば、他の指のスイッチや、ベルトやウェアに設けられたスイッチ、フットスイッチ、光スイッチ、圧力センサ、傾きセンサ等、いかなる検出器の検出結果により選択を行うようにしてもよい。スイッチ個々に割り当てられた検出対象をオン/オフする選択方法に限らず、複数スイッチの組み合わせ操作により制御したり、同じスイッチの操作で異なる制御方法を順次選択するようにしてもよい。また、操作子で選択する代わりに、同時に演奏する曲データに埋め込まれ、演奏情報とともに再生される選択イベント情報に応じて選択するようにしてもよい。このように選択された制御対象に応じて、複数画素ブロックの動きベクトルから動きベクトルを生成する方法を異ならせることにより、制御対象に応じて最適な制御を行うことができる。

【0046】また、各モードの切り換え指示を、上記実施形態のように操作子500を用いて指示してもよいし、被写体の動作によって指示できるようにしてもよい。例えば、上述のエリアモードにおいて、外部入力モードとシーケンサモードとを切り換えるためのエリアを設定しておき、当該エリアにおいて所定の動きベクトルが検出された場合はモード切り替えを行うようにしてもよい。

【0047】決定した制御量に応じて演奏を制御する手段も上記実施形態に限定されるものではなく、例えば、上記実施形態では、動きベクトルによって制御するパラメータとして、ボリューム、テンポ、パンニングを挙げたが、これらに限定されるものではなく、例えば、音色の変更や音高の変更などを行ってもよい。また、上記実施形態ではMIDIデータを用いているが、これに限らず、音楽情報をデジタルデータとして扱うことができる規格であればどのようなものでもよく、独自の規格のデータであってもよい。

【0048】上記実施形態では、プログラムはROM102に記憶されているが、これに限らず、不揮発性メモリカード、CD-ROM、フロッピーディスク、光磁気ディスク、および磁気ディスク等の可搬型の記録媒体に記録されたデータをハードディスク104等の記憶装置に転送できるように構成してもよい。このようにすれば、制御情報や制御プログラム等の追加(インストール)や更新(バージョンアップ)の際に便利である。また、可搬型の記録媒体から直接RAM103へデータを転送するようにしてもよい。さらに、可搬型の記録媒体経由ではなく、図示せぬ通信インターフェイス経由で、ハードディスク等の記録装置上の制御情報や制御プログラム等を通信ネットワーク側からダウンロードするようにしてもよい。

【0049】

【発明の効果】以上説明したように、本発明によれば、比較的簡易なシステムを用いて画像に応じた演奏の制御を行うことができる。

【図面の簡単な説明】

【図1】 実施形態の全体構成を示す図である。

【図2】 演奏制御装置の構成を示すブロック図である。

【図3】 実施形態の機能ブロック図である(その1)。

【図4】 実施形態の機能ブロック図である(その2)。

【図5】 実施形態の機能ブロック図である(その3)。

【図6】 1画面分の動きベクトルを説明する図である(その1)。

【図7】 1画面分の動きベクトルを説明する図である(その2)。

【図8】 予めエリアを設定した画像の例である。

【図9】 予め設定したエリアにおける動きベクトルを説明する図である。

【図10】 実施形態の動作のメインルーチンを示すフローチャートである。

【図11】 楽音制御処理のサブルーチンを示すフローチャートである（その1）。

【図12】 楽音制御処理のサブルーチンを示すフローチャートである（その2）。

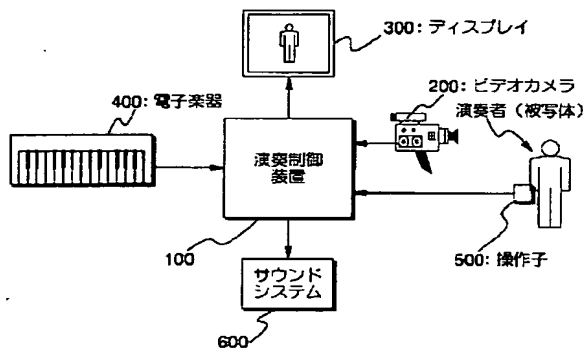
【図13】 楽音制御処理のサブルーチンを示すフロー*10

*チャートである（その3）。

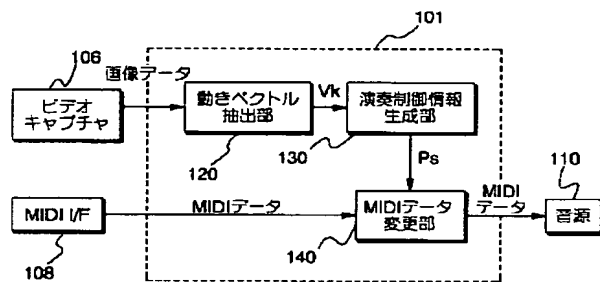
【符号の説明】

100……演奏制御装置、101……CPU、102……ROM、103……RAM、104……ハードディスク、105……タイマ、106……ビデオキャプチャ、107……ビデオ再生、108……MIDIインターフェイス、109……操作子インターフェイス、110……音源、200……ビデオカメラ、300……ディスプレイ、400……電子楽器、500……操作子、600……サウンドシステム。

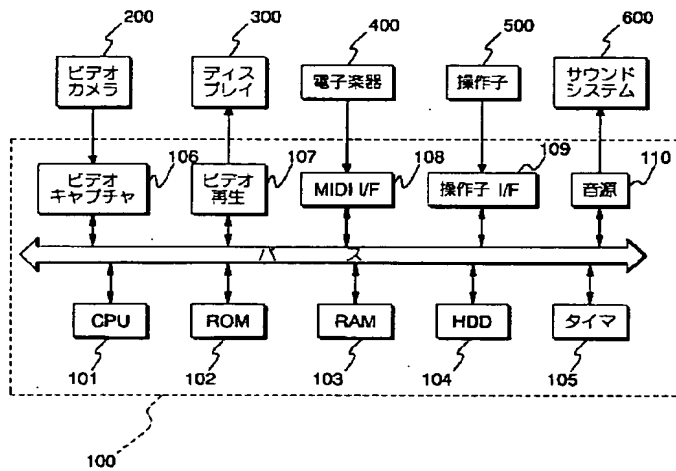
【図1】



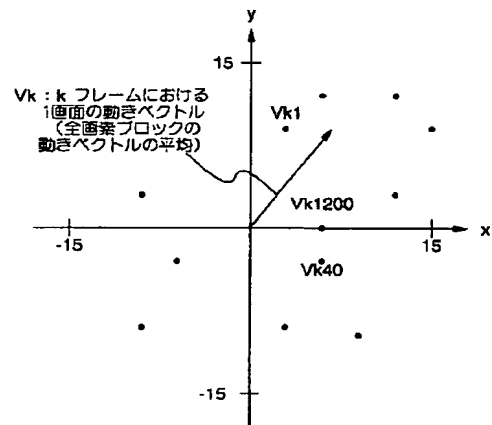
【図3】



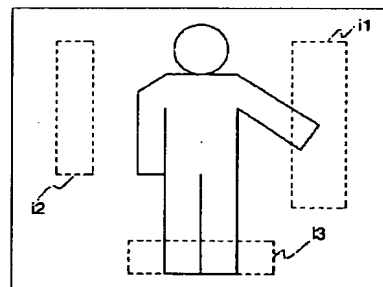
【図2】



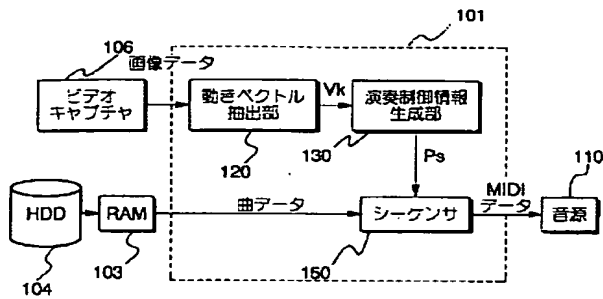
【図7】



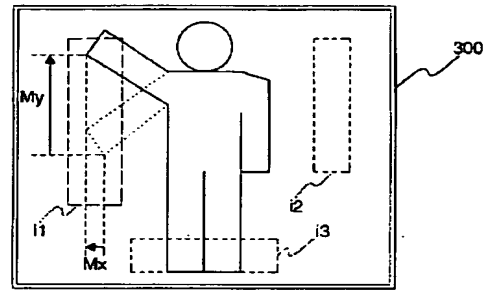
【図8】



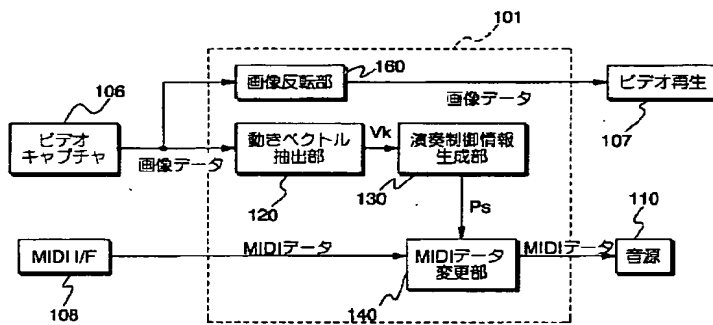
【図4】



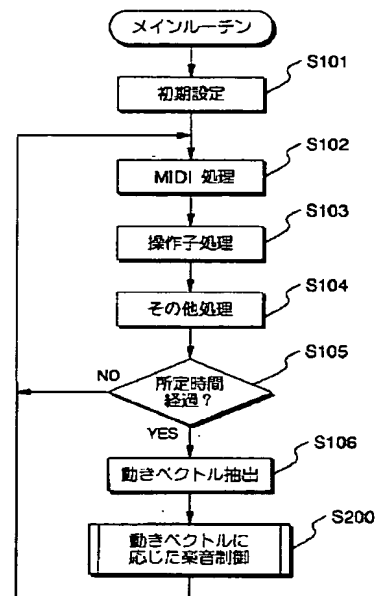
【図9】



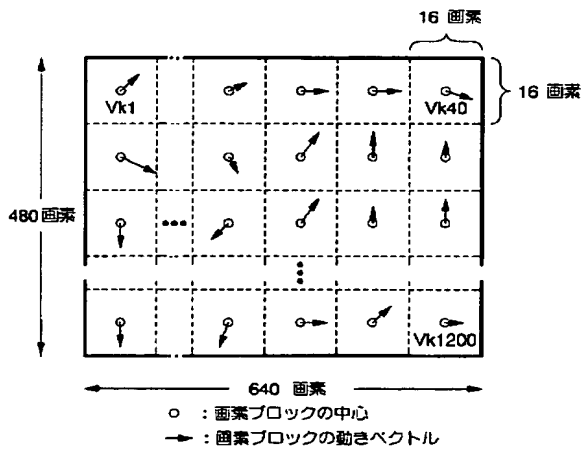
【図5】



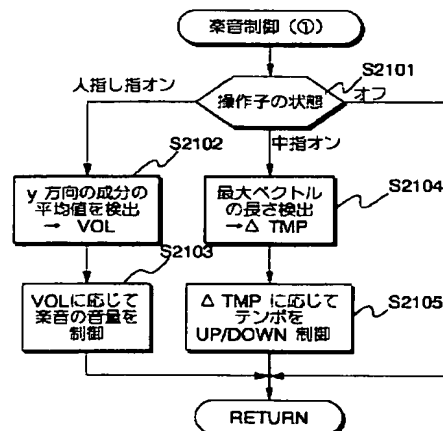
【図10】



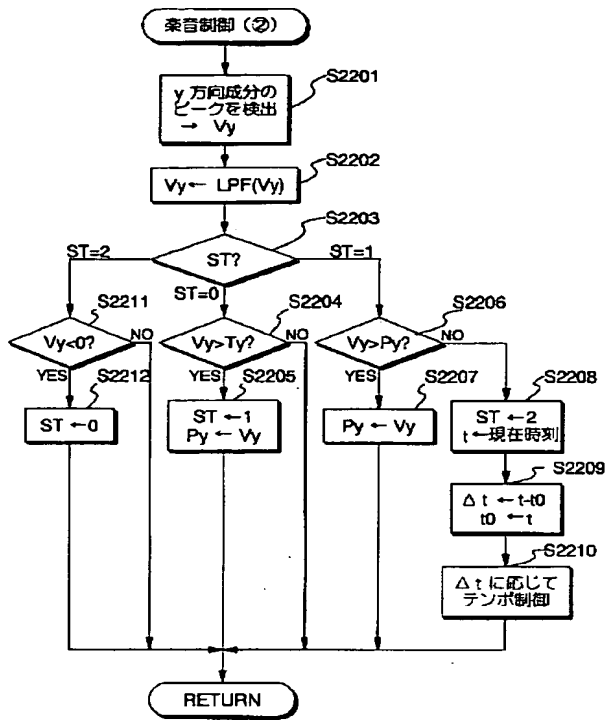
【図6】



【図11】



【図12】



【図13】

